

AI in modern RTS games

How does modern realtime-strategy games tend to implement artificial intelligence in relation to strategic decisions?

Herman Rödström

Advanced Game Project, 5SD036, 2014

Uppsala University - Campus Gotland

Tutor: Jakob Berglund Rogert

Abstract

The report discusses strategic artificial intelligence in realtime strategy games, how it can be implemented and how the project has gathered its information about the subject.

The project could not come to an answer to the posed question because it was too unspecific and the pilot study was not conducted thoroughly enough. There simply was not enough information about the specifics of how released games implement AI. Most of what was found was either general guidelines or specifics pertaining to third party bots for RTS games.

Introduction

The report will discuss the process in my work to analyze how current real-time strategy games (henceforth referred to as *RTS games*) tend to implement strategic artificial intelligence. The question formulation reads “*How does modern realtime-strategy games tend to implement artificial intelligence in relation to strategic decisions?*”. The formulation was chosen to be pertinent to the whole genre of RTS games, but at the same time not be too wide, only looking at what is deemed “*modern*” games, which in the study has been defined as a game released within the last ten years.

It is also important to note the “*strategic decisions*” part of the question, it leaves out other areas of AI, such as path finding and other low level AI which in this report is irrelevant. What the “*strategic*” part of an AI is will be discussed more thoroughly in the results section of the report.

The background and reason for this study is that I want to gain a deeper understanding of AI and how it is possible to, in the long run, improve it. I intend to keep studying this field and believe that this project can serve as a good base for future studies. To be able to study a field one must first have basic understanding of it, and also know what has already been achieved in the field.

The report details how I have been working and how I have gathered my data. In short, this has primarily been done through qualitative interviews and articles on the internet. The results will be summarized and the report will analyze my findings. It will also detail what my respondents think and believe about the subject.

During my search for good sources of information I soon realized that there was an overwhelming amount of data concerning *StarCraft 2* (Blizzard Entertainment, 2010), a very popular (Metacritic, 2014) RTS game, in comparison to other games in the genre. I have had trouble finding information about other games, which indicates that my project plan (Appendix 1) was not as thoroughly thought out as it ought to have been. If I had known beforehand that there was such an abundance of data concerning *StarCraft 2*, in relation to the other games my question formulation would have been somewhat different. After consultation with my tutor, I decided to shift my focus from RTS games in general, and instead focus more on *StarCraft 2*.

Method

As detailed in the project plan (*Appendix 1*), my main source of information have been reading articles on the web to try and find what modern RTS games actually do with their AI. Second to articles I have been conducting interviews (*Appendix 2A, Appendix 2B*), to get some insight into what others think about AI, what they think is good, or bad, and how they believe it works. I also set out to find some code guides, and to write some code myself, some small prototypes. I also aimed to conduct a quantitative survey if I found the time to do so.

As for the interviews, I conducted more interviews than I set out to do, which I see as a good thing. My respondents were all male, with some background in IT, most are, or have been, students of game design. They had varying levels of experience and expertise in RTS games but I found that all of them had enough experience to be relevant for my study. The respondents were all in their 20's, with an age difference of 7 years between the youngest and oldest.

The interviews were qualitative and conducted in a private setting. The respondents were all informed that their answers would be anonymized and that the purpose of the interview was to gather information for this study. The average time of each interview was about fifteen minutes, some required more time because the respondent had a lot of ideas, theories and thoughts about AI in RTS games. The respondents with a lot of thoughts correlated with those who claimed they had the most experience with these games. Those with less experience also had less opinions in the matter. One game in particular was discussed more often, *StarCraft 2*, as previously mentioned this has influenced the focus of this report.

The other main source of information was articles on the internet, I have been searching for keywords on search engines and websites dedicated to artificial intelligence such as AiGameDev.com. During my search for good sources I soon found that most of them were *StarCraft*, or *StarCraft 2* related, a similar experience to that I had during the interviews. I also found it a lot harder than I initially thought to find guides and articles about how the AI in specific games work. That shifted my focus towards *StarCraft 2* bots and third party code, as well as towards general guides and articles about modern AI, as opposed to articles about specific games.

I also set out to find code guides to follow and write some small prototypes with. The motivation behind this was to actually get some insight in how games actually implement AI, for some deeper knowledge. I also searched for open source AI in general, to be able to get a glance of how they work.

I left the option to conduct a quantitative survey open, in case I felt I had the time and need for it. The survey would be based upon the answers from the qualitative interviews.

Results

I found it hard to find articles and information on how specific games implement their AI, most of what I found was *StarCraft* or *StarCraft 2* related in one way or the other, as explained previously. I have some sources pertaining to RTS AI in general, but *StarCraft* and *StarCraft 2* are the only games which I have found some kind of specific information on, and what I found about them was almost exclusively about third party AIs and bots. Some interesting articles were behind a *paywall* that I could not afford, so that was very unfortunate.

Early on I found a study in *StarCraft* third party strategic AI (Ontañón et al., 2013), which is closely related to my subject. The study defines *strategic AI* to be a high-level decision making AI, in charge of the overall command, this can be compared to a general during war, the *strategic AI* makes decisions relevant to the whole team and plans for a few minutes ahead (Ontañón et al., 2013:4). This is not to be confused with a *tactical AI*, which is a commander of a smaller group of units. The *tactical AI* follows the orders of the *strategic AI*, but decides the specifics of how they should be executed individually from the *strategic AI*; it also plans for a shorter timespan, such as some tens of seconds ahead (Ontañón et al., 2013:4). This is the definition of *strategic* that is used in this report, what is relevant to the project is the overall AI and decision making, not the lower level groups of units.

The basic upkeep of a team in an RTS game, actually does not require much intelligence (*Kehoe, 2009:Strategic AI*), it is simply a balance of when and where to build what units and buildings. What requires intelligence is the interaction with players or other AIs, exploring the map to find other teams, analyzing the map for key points eg. choke points and resources. One way of sorting this information is *decision maps* (*Kehoe, 2009:Decision Maps*), which are arrays of information where each cell corresponds to part of the game map. This is then used to prioritize what units need to go to which regions.

The data from the six interviews I conducted resulted in many areas where the respondents had a consensus, and some where they disagreed. A lot of the information gathered was specific to *StarCraft 2*, which almost all of the respondents had experience with. It is important to note that those who claimed they had the most experience playing with or against AI also were those with the most opinions and thoughts on the matter. Most also played against AIs of a similar difficulty as they themselves thought they were at, experienced players played against hard difficulties and inexperienced players against easier AIs, all of them thought there was a good balance on the difficulties.

Among things that the respondents were in agreement were that regardless of the players own experience, it is always possible to set the AI to a difficulty level that suits the player. They were also in large agreement that the AI seems omnipresent, and can issue commands at several places seemingly at once, but they also thought that it was not necessarily a problem. The largest differences between the possible difficulty levels in general was that easier difficulties are slower to act and react in the game, while higher difficulty AIs act faster.

Some respondents also claimed that high difficulty AIs cheat in some games, eg. they can spawn double units or has a multiplier on the amount of resources it gains. At the same time they claimed that the easier difficulty AIs are deliberately stupid or slow, eg. not advancing to next level until the player has done so. There were mixed feelings about this, in many ways they thought it was good that the AI adapted and actually became easier or harder, but it would have been better if there was a more intelligent system to handle the difficulty.

There were very mixed feelings as to what extent the AI act convincingly like a human adversary. This was observed to the highest extent in the higher difficulties, some claimed they could almost instantly see it was an AI because it was fast, or had really recognizable patterns. Some on the other hand thought the AI in many cases behaved a lot like a human would, they said it was because it could surprise the player and make seemingly smart moves that they themselves would not think of. None of the respondents claimed that the AI always behaved like a human though, there was always some cases when it was possible to discern them from a human, regardless of what difficulty it was set on.

The respondents thought that AIs do not take risks in the same way humans do, they do careful research of their surroundings before taking a calculated risk, while humans can depend on chance at times. One key difference is also that it is almost impossible to cooperate with an AI at the same level one can with a human. The interface of communication is simply too thin, the player might be able to tell the AI to “attack here now”, but not much more than that, while humans can communicate far more complicated plans via text or audio.

When the qualitative interviews had been conducted and summarized there was not enough time to begin a quantitative survey. The qualitative interviews had provided a lot of information so it was thought that a survey would not add very much to the results.

Finding relevant code guides proved to be easier said than done. After about half of the projects defined time I realized that I had overestimated the timeframe that the project had, and needed to cut down on some of my aspirations. Part of it was also because of my problems finding good data, so I decided to focus on what I had at that time, rather than wasting time on going on with my search. I had been searching for code guides which I could follow and write some prototypes with, but found guides that felt really relevant to the project. I read some guides, but never began programming anything with them, I just scanned them briefly to see if they were relevant, but none were.

However I did find some interesting open source AI code, the code comes from a *StarCraft 2* third party AI or *bot* developed at Stanford University (Fisher, 2010). The bot is called “*Starcraft 2 Automated Player*” and it was first published in 2010, it has been in development since. This bot has a lot of informative descriptions of how it works, in addition to being open source which opens up to reviewing the actual code. I did scan through some code, which was interesting, but I did not find anything of relevance in the source code that was not already explained on the website.

Starcraft 2 Automated Player works through the same interfaces as a human in the way that it captures the actual game screen and decodes what it has captured (Fisher, 2010:Sample Frame Capture). It also sends simulated mouse and keyboard events to the game in much the same way as a human would. In this way the bot can not see all of the world map in the game, and is not omnipresent in the same way as a server-side AI can be. Even though it uses the same interfaces, it still has a lot of differences compared to a human, it is not as intelligent, but it is a lot faster, which makes up some of it not being omnipresent. This is not, however, the main field which this project is to study, but it provides some background to the rest of the bots behaviour. Since I have not found a lot of other sources of information much of this report relies on what I’ve learned from studying this bot, surely some of the solutions this bot uses is also used by other AIs.

Things such as the AI wanting to take multiple actions at one time and then having to prioritize them is something that *Starcraft 2 Automated Player* is making use of. The bot is divided up into several *threads* (Fisher, 2010:Decision Making), each in charge of an individual aspect of command. A sample of the threads are the *Army thread*, which decides whether or not to attack, retreat or scout with larger groups of units; the *Micro thread*, which controls the individual units in a group when they are in battle; the *Worker thread*, always makes sure that the *worker units* are productive and never idles. There are other threads, and the main point of this system is to divide the intelligence between different specialized units, or threads. Each thread has its own objective and makes its own decisions, they also prioritize their own goals. They each gain control over the virtual mouse and keyboard for a slice of each cycle.

Starcraft 2 Automated Player also implements a sort of memory, because in order to make intelligent decisions, an AI can not only look at the current gamestate, but has to remember what has happened previously. When trying to outsmart a human this is crucial and very complicated, but a more comprehensible example is when the *Scout thread* decides where it should venture next with its units (Fisher, 2010: *State Maintenance and Bookkeeping*). It can not only rely on what areas have been scouted thus far, but also needs to remember when each base was scouted most recently, so as to have the most recent information possible from each point of interest on the map. Because of *fog of war* the bot can not know the exact situation everywhere unless it has units observing the place, to be able to make intelligent decisions anyway, the bot needs to store relevant information such as units and buildings discovered at different locations, and when they were observed.

Analysis

I find it unfortunate that I have not found more information on how specific games implement their AI, which was the main objective of this project. The main reason for this I believe is that the games are proprietary software and it actually seems only logical that one could not so easily gain knowledge about how they work. My project plan (*Appendix I*) was of course flawed in that way, I should have realized that it would be hard to find such specific information. I did find some general information and guidelines as to how games can implement AI, and I found a lot of information on third party AIs and bots for *StarCraft* and *StarCraft 2*. This was not exactly what I set out to find but it was not completely irrelevant either. My analysis will treat what information I did find, and explain to what extent it is relevant, or not relevant.

I believe the reason I found so much about *StarCraft* and *StarCraft 2* is that they are immensely popular games, and the first installment has had tournaments and competitions (*StarCraft AI Competition, 2014*) for AI bots for years now.

The *StarCraft 2 Automated Player* has a lot of interesting features, some of which probably does not apply to how an in-game AI would use but some that probably are used by most AIs. All AIs need to remember what has happened previously in order to make smart choices, similarly to how it is implemented in *StarCraft 2 Automated Player*. However, AIs in general do not need to implement *threads* in the same way, where the intelligence is divided into sub divisions, it is possible that many do, but I have not found anything to support it and I can imagine that some AIs would use a more compressed way of dealing with things rather than dividing it.

I find it interesting that *StarCraft 2 Automated Player* uses the same interfaces as humans do, and I think that is a really good step to create better AIs in relation to they acting convincingly like humans. This, however, is most definitely not the case with most AIs, on the contrary, it is a very unique feature of *StarCraft 2 Automated Player*. Because *StarCraft 2 Automated Player* uses the same interfaces as humans it also has to remember more things than an AI with a more omnipresent role does, while the omnipresent or regular, standard AI, will look at the current gamestate in whole, *StarCraft 2 Automated Player* needs to remember the things that it does not see at the specific moment.

Some respondents to the interviews mentioned cheating AIs, the reason game developers choose this is probably because the AI did not turn out to be as smart as they wanted it to be, it seems like a cheap way to make it easier or harder, rather than developing a smarter AI they give it benefits to make it harder for the player to beat it. Respondents also had differing opinions on how much the AI behaved convincingly human, some claimed that AIs that did really smart things was a sign of human intelligence, while some claimed the same thing was a sign that a computer must have thought of this. I am not sure of how to interpret this other than that people interpret the same behaviour in different ways, and that it is really hard to create something that every player thinks is good. In general the respondents did find AIs to have a good balance and thought they are fun to play against, which must be the purpose. The problems that they saw seemed to be about how fast the AIs were and how they are omnipresent, if one would change these things, one could possibly get better AIs that people actually believed.

As previously explained I did not find any good guides, or a lot of code examples, and in retrospect I don't think that guides would really benefit the project very much, and it seems like I should have left that out of the original project plan (*Appendix I*). On a personal level I would most probably gain from following through with some guides, but in the case of this project I think that some more articles more pertinent to my question would benefit the project better than guides in the same field.

I had a lot of trouble finding relevant information, and I undoubtedly failed to answer the question I posed in the beginning, it was simply too wide and unspecific. I also should have realized that acquiring information about specific games implementations of AI would be too hard. In spite of this I feel that I have learned about how strategic AI works, even though the project would benefit from more specific information and more time to research. As for the problem with *paywalls*, a future project on the subject would probably benefit from a small budget that would make those articles available. Better planning of my time and more importantly, a more thought through pilot study would also have helped. A better pilot study would have narrowed the spectrum of the project and would not have set out to do more things than were needed or that there was time for.

Conclusion

The project has not really arrived at an answer to the question posed in the beginning. It was too wide and the pilot study was not made as thoroughly as it should have. The project has, however, gained knowledge as to how AIs can be implemented even though much of what has been found are special cases, and most are third party bots. A very large portion of the data is related to *StarCraft* or *StarCraft 2*, as well as a lot of general AI guidelines, rather than specific information about a few select games.

References

Papers

Ontañón, S., Synnaeve, G., Uriarte, A., Richoux, F., Churchill, D. and Preuss, M.. 2013. “*A Survey of Real-Time Strategy Game AI Research and Competition in StarCraft*”. IEEE Transactions on Computational Intelligence and AI in Games (volume 5, issue 4, pages 293 - 311). https://web.archive.org/web/20140401200906/http://webdocs.cs.ualberta.ca/~cdavid/pdf/starcraft_survey.pdf (Accessed 2014-04-01). doi:10.1109/TCIAIG.2013.2286295

Websites

Fisher, Matt. 2010. “*Starcraft 2 Automated Player*”. Stanford University - Matt’s Webcorner. <https://web.archive.org/web/20131004114632/http://graphics.stanford.edu/~mdfisher/GameAIs.html> (Accessed 2013-10-04).

Kehoe, Donald. 2009. “*Designing Artificial Intelligence for Games (Part 3: Tactical & Strategic AI)*”. Intel Developer Zone. <https://web.archive.org/web/20121101085842/http://software.intel.com/en-us/articles/designing-artificial-intelligence-for-games-part-3/> (Accessed 2012-11-01).

Metacritic. 2014. *PC Game Releases by Score, All time*. <https://web.archive.org/web/20140108083042/http://www.metacritic.com/browse/games/score/metascore/all/pc?view=condensed&sort=desc> (Accessed 2014-01-08).

StarCraft AI Competition. 2014. “*StarCraft AI Competition*”. <http://webdocs.cs.ualberta.ca/~cdavid/starcraftaicomp/> (Accessed 2014-04-03).

Games

Blizzard Entertainment. 2010. *StarCraft 2 - Wings of Liberty*. <http://eu.blizzard.com/en-gb/games/sc2/>

Appendices

Appendix 1: The project plan detailing the pupose and method of this project. *(Swedish language)*

Appendix 2A: A breakdown of the interview results from this project. *(Swedish language)*

Appendix 2B: The raw data, written short, from the interviews. *(Swedish language)*